

Introdução ao Desenvolvimento de Jogos: Do Protótipo ao Protótipo Jogável

Informações Gerais

Carga Horária: 40 horas

Público-Alvo: Iniciantes em desenvolvimento de jogos com conhecimentos básicos de Python

Pré-requisitos: Python básico (variáveis, loops, funções, classes)

Framework: Pygame

Objetivo Geral

Capacitar os participantes a desenvolver protótipos jogáveis de jogos digitais, compreendendo todo o processo desde a concepção da ideia até a implementação de mecânicas funcionais, utilizando ferramentas modernas e boas práticas de desenvolvimento.

Objetivos Específicos

- Compreender os fundamentos do design de jogos e prototipagem
- Dominar conceitos essenciais de game loops e mecânicas de jogo
- Implementar sistemas básicos de física, colisão e controle de personagens
- Desenvolver interfaces de usuário funcionais
- Aplicar princípios de game feel e feedback ao jogador
- Publicar e testar protótipos com usuários reais

Conteúdo Programático

Módulo 1: Fundamentos do Game Design (8h)

1.1 Introdução ao Desenvolvimento de Jogos

- Pilares do game design: mecânicas, narrativa, estética e tecnologia
- Análise de jogos: desconstruindo mecânicas populares
- O que é um protótipo e por que prototipar?

1.2 Concepção e Documentação

- Brainstorming de ideias: técnicas criativas
- Game Design Document (GDD) simplificado
- Definição de escopo para protótipos

- Core gameplay loop: o coração do seu jogo

1.3 Introdução ao Pygame

- O que é Pygame e suas vantagens
- Instalação e configuração do ambiente (pip install pygame)
- Estrutura básica: janela, loop principal, eventos
- Sistema de coordenadas e renderização

Módulo 2: Prototipagem Rápida (10h)

2.1 Primeiros Passos com Pygame

- Display e Surface: criando e manipulando telas
- Game loop: update e draw
- Tratamento de eventos (QUIT, KEYDOWN, MOUSEBUTTONDOWN)
- Clock e controle de FPS

2.2 Implementando Mecânicas Core

- Classes para GameObject: Player, Enemy, Item
- Input handling: pygame.key.get_pressed()
- Movimentação básica com delta time
- Sprites e pygame.image.load()

2.3 Game Loop Básico

- Estados do jogo (menu, jogando, pause, game over)
- Sistema de pontuação simples
- Condições de vitória e derrota

Prática: Desenvolvimento de um protótipo minimal (ex: endless runner simples)

Módulo 3: Mecânicas e Sistemas (10h)

3.1 Física e Colisões

- pygame.Rect: detecção de colisão retangular
- colliderect() e collidelist()
- Física simples: velocidade, aceleração, gravidade
- Implementação de plataformas e pulo

3.2 Sistemas de Combate/Interação

- Sistema de vida e dano
- Sprites Groups para organizar objetos
- Projéteis com `pygame.sprite.Sprite`
- Inimigos com IA básica (patrulha, perseguição)

3.3 Level Design Básico

- Princípios de level design
- Tilemaps e construção de cenários
- Spawn points e checkpoints

Prática: Expansão do protótipo com inimigos e objetivos

Módulo 4: Polish e Game Feel (8h)

4.1 Interface e Feedback Visual

- `pygame.font`: renderizando texto
- HUD: vida, pontuação, minimapa
- `pygame.draw` para formas e efeitos
- Partículas simples com listas de objetos

4.2 Áudio e Feedback Sonoro

- `pygame.mixer`: carregando sons e música
- Sound effects (SFX) com `pygame.mixer.Sound()`
- Música de fundo com `pygame.mixer.music`
- Importância do áudio no game feel

4.3 Game Feel e Juice

- Screen shake e impacto
- Tweening e animações procedurais
- Timing e weight nas animações

Prática: Polimento do protótipo com feedback audiovisual

Módulo 5: Finalização e Publicação (4h)

5.1 Testes e Iteração

- Playtesting: como testar seu jogo
- Coleta e análise de feedback

- Iteração sobre o protótipo

5.2 Build e Exportação

- Configurações de build para diferentes plataformas
- Otimização básica
- Debugging e correção de bugs críticos

5.3 Publicação do Protótipo

- Plataformas para protótipos (itch.io, GameJolt)
- Apresentação do projeto: screenshots e descrição
- Próximos passos: do protótipo ao jogo completo

Prática: Publicação do protótipo final em plataforma online

Metodologia

- **40% Teórico:** Exposição de conceitos, análise de jogos, discussões
- **60% Prático:** Desenvolvimento hands-on de protótipos, exercícios guiados
- Aprendizagem baseada em projetos
- Feedback contínuo e iterativo
- Sessões de playtesting em grupo

Avaliação

- **Participação e exercícios práticos:** 30%
- **Protótipo intermediário (Módulo 3):** 30%
- **Protótipo final jogável:** 40%

Recursos Necessários

- Python 3.8+ instalado
- Pygame instalado (pip install pygame)
- Editor de código (VS Code, PyCharm, ou similar)
- Assets gratuitos (serão fornecidos links)
- Conta em plataforma de publicação (itch.io - gratuita)
- **50 KitKats para dinâmicas gamificadas!**

Resultados Esperados

Ao final do curso, o aluno será capaz de:

- Criar protótipos jogáveis funcionais
- Implementar mecânicas básicas de jogos 2D ou 3D simples
- Aplicar princípios de game design e game feel
- Publicar e compartilhar seus protótipos online
- Continuar aprendendo de forma autônoma

Bibliografia Recomendada

Livros:

- "The Art of Game Design: A Book of Lenses" - Jesse Schell
- "Making Games with Python & Pygame" - Al Sweigart (gratuito online)
- "Game Programming Patterns" - Robert Nystrom

Online:

- Documentação oficial do Pygame (pygame.org)
- Game Maker's Toolkit (YouTube)
- Clear Code - Pygame Tutorials (YouTube)
- GDC Talks (YouTube)

Observações

- O curso é focado em prototipagem com Pygame (Python)
- Flexibilidade para escolha de gênero de jogo pelos alunos
- Foco em funcionalidade sobre gráficos sofisticados
- Código limpo, comentado e boas práticas serão priorizados
- Dinâmicas gamificadas com recompensas (KitKats!)

Dinâmicas Gamificadas com KitKats

1. Bug Bounty Hunter

- Quando um aluno encontra um bug no código de exemplo ou no jogo de outro colega durante playtesting: **1 KitKat**
- Bônus se explicar como corrigir: **+1 KitKat**

2. Code Review Champion

- Aluno que faz a melhor pergunta ou sugestão durante code review em grupo: **2 KitKats**
- Votação rápida da turma ao final da sessão

3. Checkpoint Achievements

- Primeira janela Pygame funcionando: **1 KitKat**
- Primeiro sprite movendo: **1 KitKat**
- Primeira colisão detectada: **1 KitKat**
- Primeiro inimigo com IA: **2 KitKats**
- Jogo com som funcionando: **1 KitKat**

4. Creative Twist Award

- Aluno que implementar a mecânica mais criativa/diferente no protótipo intermediário: **3 KitKats**
- Votação da turma

5. Help Desk Hero

- Aluno que ajudar outro colega a resolver um problema técnico (validado pelo instrutor): **1 KitKat**
- Incentiva colaboração

6. Speed Coding Challenge

- Desafios rápidos (10-15 min): "implemente um pulo", "faça a bola quicar"
- Primeiro a completar corretamente: **2 KitKats**
- Pode fazer 3-4 ao longo do curso

7. Best Game Feel

- Protótipo com melhor sensação de controle/feedback (votação no módulo 4): **5 KitKats**

8. Participation Points

- Responder perguntas durante aulas teóricas: **1 KitKat**
- Fazer perguntas relevantes: **1 KitKat**
- Limite de 1 por dia por aluno para distribuir

9. Clean Code Award

- Melhor código comentado e organizado (revisar no módulo 3): **3 KitKats**

10. Playtester MVP

- Aluno que der o feedback mais construtivo durante sessões de teste: **2 KitKats**

11. Final Boss: Launch Day

- Todos que publicarem o protótipo final no prazo: **2 KitKats garantidos**
- Projeto mais completo: **5 KitKats extras**

Regras das Dinâmicas:

- Total de ~50 KitKats distribuídos ao longo do curso
- Clima leve e divertido, não competitivo demais
- Instrutor tem "joker" para dar KitKats por esforço/persistência
- Pode trocar KitKats entre alunos (economia interna do curso!)

Dica: Anuncie as dinâmicas gradualmente para manter o elemento surpresa!